

Docket No.: 50253-118 (P2287)

PATENT

AF  
2731  
2631  
#11/28/00  
D

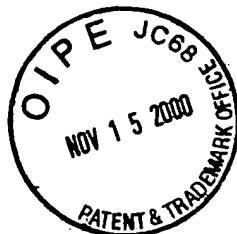
IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of

AMIT GUPTA, et al.

Serial No.: 08/868,972

Filed: June 04, 1997



Group Art Unit: 2731

Examiner: Brenda Pham

RECEIVED

NOV 17 2000

Technology Center 2600

For: TECHNIQUES FOR IMPROVING VIRTUAL CHANNEL MANAGEMENT AND  
MAINTENANCE IN A NETWORK ENVIRONMENT

**REQUEST FOR REINSTATEMENT OF THE APPEAL**  
**AND TRANSMITTAL OF SUPPLEMENTAL APPEAL BRIEF UNDER 37 CFR § 1.193(B)(2)**

Assistant Commissioner for Patents  
Washington, DC 20231

Sir:

Reinstatement of the Appeal, for which a Notice of Appeal was filed March 23, 2000, is hereby requested under 37 CFR 1.193(b)(2).

Submitted herewith in triplicate is a Supplemental Appeal Brief in support of the Appeal with respect to rejected claims 1, 2, 6-19, and 21-30.

It is requested that earlier paid fees for the Notice of Appeal and Appeal Brief be applied to the present Appeal as provided in MPEP § 1208.02.

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 500417 and please credit any excess fees to such deposit account.

Respectfully submitted,

MCDERMOTT, WILL & EMERY



Thomas D. Robbins

Registration No. 43,369

600 13<sup>th</sup> Street, N.W.  
Washington, DC 20005-3096  
(202) 756-8000 TDR:daf  
**Date: November 15, 2000**  
Facsimile: (202) 756-8087

Docket No.: 50253-118 (P2287)

PATENT

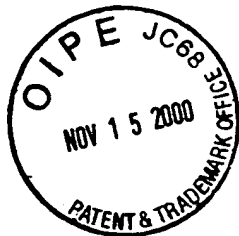
**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of

AMIT GUPTA, et al.

Serial No.: 08/868,972

Filed: June 04, 1997



Group Art Unit: 2731

Examiner: Brenda Pham

For: TECHNIQUES FOR IMPROVING VIRTUAL CHANNEL MANAGEMENT AND  
MAINTENANCE IN A NETWORK ENVIRONMENT

**RECEIVED**

**NOV 17 2000**

**Technology Center 2600**

**SUPPLEMENTAL APPEAL BRIEF**

Assistant Commissioner for Patents  
Washington, DC 20231

Sir:

This Supplemental Appeal Brief is submitted in support of the Notice of Appeal filed March 23, 2000, and in response to the Office dated August 15, 2000 rejecting claims 1, 2, 6-19, and 21-30.

**REAL PARTY IN INTEREST**

The real party in interest is SUN MICROSYSTEMS INC. of Mountain View, California.

**RELATED APPEALS AND INTERFERENCES**

There are no related appeals or interferences that will affect or be affected by the decision in this case.

**STATUS OF CLAIMS**

Claims 1-30 remain in the application. Claims 1, 2, 6-19, and 21-30 stand rejected. Claims 3-5 and 20 stand objected to. The independent claims are 1, 10-11, 17-18, 21 and 23-26.

### STATUS OF AMENDMENTS

No amendments have been filed subsequent to the last Office Action.

### SUMMARY OF INVENTION

A digital communications network consists of nodes that are connected by physical links. A user is located at a terminal which is connected by a link to a node. Each link enters or leaves a node at a physical port of the node. Each node typically includes many input ports and many output ports, a switch that can be operated to connect any input port to any output port, and a control element or control point to operate the switch. At each node a routing process executes to determine which node is next on a route from the originating node to the destination node and which output port(s) are associated with the next node. The routing process uses a destination identification, contained typically in a packet, for information received in on a particular input port to determine the particular output port to use. The results of the routing process determination is an input port virtual circuit to output port virtual circuit association, perhaps stored in a look-up table or other data structure. The control point uses the results of the routing process to connect the particular input port to the particular output port when a packet arrives.

Rather than a physical connection, a virtual connection, sometimes called a virtual channel or virtual circuit (VC) is defined between the source of the packet and its destination. A virtual circuit gives the appearance of maintaining a hardwire connection, but utilizes the resources of the connection only when data needs to be sent. This permits the link to be shared by other virtual circuits and improves the efficiency and throughput of the links of the network.

Before a virtual circuit (VC) can be used, as described above, it must be set up in a separate set of transactions. When the communications session is terminated, the virtual circuit must be broken

down, also called "taken down" or "torn down" in another set of transactions. Even when packets are not being transmitted, the VC information has to be maintained.

In some networks, once a series of nodes are used to connect a first user at a first terminal to a second user at a second terminal, the VCs are maintained until all the information involved in the communications between the first and second user has passed and one or both users terminate the communications session.

A particular virtual circuit represents a set of associations between a particular origination user terminal and a particular destination user terminal. The node switch includes a mechanism to associate the particular virtual circuit with its input and output ports. When information arrives on the input port indicating a virtual circuit, the switch finds the associated output port. A matching path is generally maintained only until the information in the packet has traversed the switch to the output port. The next packet arriving at a particular input port may be associated with a different virtual circuit and thus require a different path. Switching required to accommodate packets associated with different virtual circuits can place a great load on resources available at a node. Sometimes many millions of switching operations per second may be required.

With conventional VCs, a separate VC is set up for each pair of origination user and destination user. The set up transactions include the origination user sending a message to a first node connected to the origination user's terminal. The message requests a connection to the destination user. At the first node, as shown in Figure 4 of the specification, the routing process is executed to determine which of the other nodes connected to the first node should be the next node to get the message requesting the connection. A port linking the first node to the determined next node is identified and stored at the first node with a locally generated virtual circuit identification (VCI), and then the message is forwarded with the VCI to the next node. This process is repeated at all the next nodes, each using locally

generated VCIs, until a node is reached that is connected to the terminal of the destination user. If a connection can be established (e.g., the terminal is neither busy nor off) the last node sends an acknowledgment to the preceding node with the VCI sent by the preceding node, and that acknowledgment propagates back over the switch chain. Tables or other data structures are updated at the last node to identify the virtual circuit associated with the VCI. Tables are also updated at intermediate nodes to identify the virtual circuit.

Unfortunately, if many communicating sessions are needed, a corresponding number of virtual circuits are needed to carry all the information, and one must repeat the laborious set up and tear down process for each of the virtual circuits at each of the intervening nodes. To reduce the overhead of such repeated operations for essentially the same desired connection between origination node and destination node, permanent virtual circuits have been used. Permanent virtual circuits are those that are set up and remain established for several different communications sessions before being broken down. However, each permanent virtual circuit still must be established individually.

Virtual circuits are named locally by the node that requests them and can be represented by an ID selected from a small set of identifiers. The concept of a virtual path was promoted as part of the Asynchronous Transfer Mode (ATM) protocol to use the same name throughout the network for the same virtual circuit. A virtual path can contain up to  $2^{16}$  (65,536) virtual circuits to uniquely identify all the virtual circuits in a network. Thus a group of VCs can be included in a path. However, 16 bits are needed to specify the virtual circuit within the virtual path. These 16 bits require hardware changes in the switches at the nodes.

The techniques of the present invention allow virtual circuits to be set up and broken down in fewer switch operations than conventional virtual circuits by processing (e.g., setting up, breaking down, and maintaining) a group of virtual circuits together in a virtual circuit bunch (VCB) that does

not require the additional hardware of virtual circuit paths. The various characteristics of a VCB are described in the specification. A VCB "establishes a plurality of virtual circuits from one node to at least one other node.....in response to a single request" (specification, page 4, lines 22-25). "[A] group of virtual circuits [is] pre-established in a virtual circuit bunch" (VCB) (specification, page 23, lines 19-20). As described in the specification "a hierarchy of aliases . . . [is] utilized to conveniently refer to portions of . . . a virtual circuit bunch," (specification, page 23, lines 17-20).

The advantage of such an arrangement is that "Virtual Circuit Bunches [are] utilized to set-up and manage, together, groups of virtual circuits in a flexible way which results in increased performance as well as overcoming the problems of the prior art." (Specification, page 4, lines 5-8.) The virtual circuit bunch (VCB) "enables groups of virtual circuits to be established . . . without any changes to switch hardware" (specification, page 4, lines 10-13).

As an example of the advantages of a VCB, consider the set up of a VCB for communications between node 1 and nodes 3, 4 and 5 described in the specification starting on page 19 with reference to Figure 7C. In this example, an ATM network is used in which nodes conventionally maintain tables for each input port that associates a local virtual circuit ID (VCI) with an output port, and other tables for each output port that associates a VCI with an input port. The tables and processing are different when a virtual circuit bunch (VCB) is used.

According to this example of the present invention, a packet arrives requesting set up of a VCB. The packet includes an identification of several destinations, specifically, node 4, node 5 and node 3. For each destination, the packet indicates a number of virtual circuits needed, specifically, 8, 8 and 9, respectively, and indicates a level of service needed. The packet of Figure 7C is a single request message that is used to generate 25 virtual circuits from switch 1 to switches 3, 4 and 5. This request

message is received at node 2 and is used by node 2 to set up 25 bi-directional virtual circuits (called VCB 3) with node 1.

Assuming that node 2 is physically linked with nodes 3, 4 and 5, then node 2 makes table entries that associate 8 virtual circuits (called VCB 3A) of the 25 virtual circuits VCB 3 with the ports to links going to node 4. Likewise, node 2 makes table entries that associate 8 other virtual circuits (called VCB 3B) of the VCB 3 virtual circuits with ports to links going to node 5. Finally, node 2 makes table entries that associate 9 virtual circuits (called VCB 3C) of the VCB 3 virtual circuits with ports to links going to node 3. Node 2 does all this based on the single request of Figure 7C. Then node 2 sends a single request message to node 4 requesting 8 virtual circuits in VCB 3A. Node 2 sends a similar request to node 5 and another to node 3. Thus four requests are sent during the establishment of this VCB. Using conventional virtual circuits, 50 requests would be needed, 25 requests to node 2, 8 requests to node 4, 8 more requests to node 5 and 9 more requests to node 3.

Node 2 then receives three acknowledgement messages, one each from nodes 4, 5 and 3, respectively. Then node 2 sends a single acknowledgement message to node 1 indicating all 25 virtual circuits in the request have been established. Using conventional virtual circuits, 50 acknowledgements would be needed, 25 acknowledgments from node 2, 8 acknowledgment from node 4, 8 more acknowledgment from node 5 and 9 more acknowledgment from node 3. Counting both request and acknowledgement messages, a VCB is set up with 6 messages where conventional virtual circuits require 100 messages. This represents a tremendous performance advantage. Note that, the six messages of this example are generated in response to a single request (the first message) to set up the VCB sent by node 1.

Note also that only the 25 needed virtual circuits are specified in the requests. Thus any short name or code that can distinguish 25 different virtual circuits can be used as a name or alias for these



virtual circuits. Four bits are sufficient to distinguish up to 32 different items (i.e., 0 to 31, inclusive). There is no need to use a 16-bit naming convention for the virtual circuits in the bunch and thus no need for the hardware additions required for "virtual circuit paths."

After setup, whenever a communications session is needed between a first user at node 1 with a second user at one of the destinations of the VCB, say node 5, then node 2 uses its tables to associate the arriving data message, which specifies destination node 5, with one of the virtual circuits in VCB 3B that is not already being used. Then the control point at node 2 makes an entry into a table of virtual circuits being used by a given input port, sets the switch to connect to the associated output port, and sends the data message on to node 5. The repeated setup and breakdown of individual virtual circuits is eliminated without resorting to virtual circuit paths and the additional hardware that they require.

Other aspects of the invention deal with breaking down virtual circuit bunches, refreshing them, sharing them for fast connect services, split routing of a virtual circuit bunch, interleaving conflicts, and aggregating virtual circuits into existing virtual circuit bunches, among other techniques.

### ISSUES

The issues on appeal are:

- A. Whether the Examiner erred in rejecting claims 1, 2, 6, 7, 10, 11, 19, 21, and 22 under 35 U.S.C. §102(b) as being anticipated by Subramanian et al., U.S. Patent 5,519,707 (Subramanian).
- B. Whether the Examiner erred in rejecting claim 17 under 35 U.S.C. §102(b) as being anticipated by Fisk, U.S. Patent 5,274,643.
- C. Whether the Examiner erred in rejecting claims 8, 12, and 18 under 35 U.S.C. §103(a) as being unpatentable over Subramanian.

D. Whether the Examiner erred in rejecting claim 9 under 35 U.S.C. §103(a) as being unpatentable over Subramanian in view of Suzuki, U.S. Patent 4,884,263.

E. Whether the Examiner erred in rejecting claims 14, 15, and 24 under 35 U.S.C. §103(a) as being unpatentable over Subramanian in view of Fisk.

F. Whether the Examiner erred in rejecting claims 13 and 26 under 35 U.S.C. §103(a) as being unpatentable Subramanian in view of Hiller.

#### GROUPING OF CLAIMS

All claims are argued separately, and each claim stands or falls independently of any other claim; except, claim 2 stands or falls with claim 1, claim 7 stands or falls with claim 6, and claim 19 stands or falls with claim 18.

#### THE ARGUMENT

A. The Examiner erred in rejecting claims 1, 2, 6, 7, 10, 11, 19, 21-23, and 27 under 35 U.S.C. §102(b) as being anticipated by Subramanian

Claim 1 recites *inter alia*, a "controller configured to set up at least one group of virtual circuits . . . as a virtual circuit bunch."

Similarly, claim 10 recites *inter alia*, a "processor, connected to said bus, said processor configured to . . . generate a single request to said switching node to establish a plurality of virtual circuits to respective one or more destinations as a virtual circuit bunch."

Similarly, claim 11 recites *inter alia*, "establishing a plurality of virtual circuits . . . as a virtual circuit bunch."

Similarly, claim 21 recites *inter alia*, "at least one of said node controllers is configured to set up a group of virtual circuits . . . as a virtual circuit bunch."

Similarly, claim 22 recites *inter alia*, a "controller configured to set up at least one group of virtual circuits . . . as a virtual circuit bunch."

Similarly, claim 23 recites *inter alia*, a "computer program comprising instructions for establishing a plurality of virtual circuits . . . as a virtual circuit bunch."

As is clear from the specification, and explained in the summary of the invention above, a virtual circuit bunch (VCB) is different from a group of virtual circuits, and is different from permanent virtual circuits, and is different from a virtual path. One characteristic of a virtual circuit bunch is that all the virtual circuits in the bunch are established as a result of a single request to set up the bunch. Another characteristic of a VCB is that all the virtual circuits in the VCB can be identified with few enough bits that additional hardware changes in a switch are not required.

Subramanian is directed to setting up virtual paths between a central management supervisor 202 and each of a set of switches in a network. (Fig. 6). On such a logical star arrangement, virtual circuit identifiers can be named by service type 704 and port ID (Fig. 7B). Subramanian uses permanent virtual paths as known in the prior art, but requests those paths from the central management supervisor 202. Specifically, Submaranian teaches a first "virtual service path [is] established between the first switch and the supervisor. Likewise, a second . . . virtual service path [is] established between the second [switch] and the supervisor" (Subramanian, column 3, lines 47-52). No details on setting up the virtual paths are given, so no suggestion is made that these virtual paths differ from the known prior art paths. Submaranian then teaches that "channel numbers within each service path are preassigned by the supervisor" (Subramanian, column 3, lines 55-56). Thus virtual circuits are not set up together by the switch controllers themselves in bunches, as in Appellant's invention, but

instead are set up in conventional ways by the supervisor. There is no suggestion or teaching that the virtual circuits of a virtual path are set up together in response to a single request as in a VCB.

Because Subramanian does not teach or suggest the VCB of claims 1, 10, 11, and 21-23, the rejection of claims 1, 10, 11, and 21-23 under 35 U.S.C. §102(b) is improper. For at least the same reasons, the rejection is improper for claims 2, 6, and 7, which depend, directly or indirectly, from claim 1. For at least the same reasons, the rejection is improper for claim 22, which directly depends from claim 21. For at least the same reasons, the rejection is improper for claim 27, which directly depends from claim 23.

In addition, claim 6 recites "one of a plurality of virtual circuits of a virtual circuit bunch" which is not shown by any of the references. Since Subramanian fails to show a VCB, Subramanian cannot show a virtual circuit of a VCB and cannot show assigning digital information to such a virtual circuit of a VCB.

Claim 7 depends from claim 1 which recites "a virtual circuit bunch" and "a controller configured to set up at least one group of virtual circuits . . . as a virtual circuit bunch." Neither limitation is taught or suggested by or Subramanian for the reasons given above.

Claim 19 depends from claim 18 which recites "a virtual circuit bunch" which is not taught or suggested by Subramanian for the reasons given above. Because Subramanian does not teach or suggest every limitation of Appellants' claim, a rejection of claim 18 would be improper. Since such a rejection was not made, Appellants presume that the Examiner recognizes that a rejection of claim 18 as being anticipated by Subramanian within the meaning of 35 U.S.C. § 102(b) would be improper. As further evidence that Examiner recognizes that a rejection of claim 18 as being anticipated by Subramanian within the meaning of 35 U.S.C. § 102(b) would be improper, claim 18 was rejected under 35 U.S.C. § 103(a), the error of which will be described below in section C. Since claim 19

depends from claim 18, and therefore includes all the limitations thereof, the rejection is also improper with respect to claim 19.

In addition, claim 22 recites "virtual circuit . . . is connected . . . using . . . said virtual circuit bunch" which are not shown by Subramanian for the reasons given above.

In addition, claim 27 recites "transmitting said instructions . . . to a destination over a communications interface" which is not shown in Subramanian. The Examiner does not address whether Subramanian shows instructions are transmitted to a destination over a communications interface. Furthermore, since Subramanian does not show instructions for establishing a plurality of virtual circuits as a VDB, it does not, and cannot, show transmitting such instructions.

For the reasons given, it is submitted that the Examiner's rejection of claims 1, 2, 6, 7, 10, 11, 16, 19, 21-23, and 27 under 35 U.S.C. §102(b) as being anticipated by Subramanian is untenable. Accordingly, Appellants respectfully request reversal of the rejection.

B. The Examiner erred in rejecting claim 17 under 35 U.S.C. §102(b) as being anticipated by Fisk

Claim 17 recites *inter alia*, "aggregating those virtual circuits into a virtual circuit bunch."

As described above in section A, and in the specification, the difference between a virtual circuit bunch (VCB) and 1) a group of virtual circuits, 2) permanent virtual circuits, and 3) a virtual path has been established.

The Examiner asserts that Fisk teaches a VCB, citing a passage which states "[t]he criteria for determining whether the virtual circuit can be grouped is dependent on the network node criteria" (Fisk, column 5, lines 38-40). However, as made clear in the specification, a virtual path is not a VCB; therefore the virtual paths of Fisk do not teach or suggest a VCB. There is no reason why one of

ordinary skill in the art would conclude that the virtual paths of Fisk differ from the virtual path concept of the ATM protocol that requires additional hardware. For example, Fisk notes that "telecommunication equipment changes . . . In particular telecommunication manufacturer StrataCom . . . has developed network communication modules . . . that groups . . . "virtual circuits" having like characteristics [into] virtual paths" (Fisk, column 1, lines 50-57).

There is nothing in Fisk that shows the virtual paths have any of the characteristics of a VCB. Fisk does not teach how the virtual paths are set up or maintained or broken down. In fact Fisk is not even related to the details of actual, real-time establishment and use of virtual circuits. Instead Fisk is directed to simulating overall network performance in order to design network topology, i.e., the number and connections of nodes in the network. For example, Fisk states that it is a an object of the Fisk invention "to provide a network design tool which takes into account the routing of virtual circuits over virtual paths" and "to provide a cost measure" (Fisk, column 2, lines 15-19). Appellants submit that Fisk assumes virtual paths are known, and uses the known properties of virtual paths in a network design tool. Fisk does not even suggest that virtual paths differ from those known and described in the specification of Appellants' application.

For the reasons given, the Examiner's rejection of claim 17 under 35 U.S.C. §102(b) as being anticipated by Fisk should be reversed. Accordingly, Appellants respectfully request such action.

- C. The Examiner erred in rejecting claims 8, 12, 18, 25 and 29 under 35 U.S.C. §103(a) as being unpatentable over Subramanian

Claim 8 depends from claim 1 which recites *inter alia*, "a virtual circuit bunch" and " a controller configured to set up at least one group of virtual circuits . . . as a virtual circuit bunch." Neither limitation is taught or suggested by or Subramanian for the reasons given above in section A.

In addition, claim 8 recites *inter alia*, "virtual circuits of a virtual circuit bunch going to a single destination . . . routed over different paths" which is not shown by Subramanian. Since Subramanian does not show a VCB, it cannot show virtual circuits of a VCB being routed differently.

Claim 12 depends from claim 11, which recites *inter alia*, "establishing a plurality of virtual circuits . . . as a virtual circuit bunch." At least this limitation is not taught or suggested by Subramanian for the reasons given above in section A.

In addition, claim 12 recites "setting up switching tables when at least one subsequent node has acknowledged." This is not taught or suggested by the references as the Examiner admits (Office Action of August 15, 2000, page 5). The Examiner argues that a switching table is updated when an acknowledgement is received, but the Examiner does not show where a plurality of "tables" can be set up upon the receipt of "one" acknowledgement from a node, as required by claim 12.

Independent method claim 18 recites *inter alia*, "assigning a packet to a virtual circuit of a virtual circuit bunch" which is not shown by Subramanian. Subramanian does not teach or suggest a VCB and thus cannot assign a packet to a virtual circuit selected from a VCB. Thus the rejection of claim 18 is improper.

Similarly, independent computer program product claim 25 recites "assigning a packet to a virtual circuit of a virtual circuit bunch" which is not shown by Subramanian. Subramanian does not teach or suggest a VCB and thus cannot assign a packet to a virtual circuit selected from a VCB. Thus the rejection of claim 25 is improper. For at least the same reasons, the rejection is improper for claim 29, which depends directly from claim 25.

→ In addition, claim 29 recites "transmitting said instructions . . . to a destination over a communications interface" which is not shown by Subramanian. The Examiner does not address whether Subramanian teaches that instructions to assign a packet are transmitted to a destination over a

communications interface. Furthermore, since Subramanian does not show instructions for assigning a packet to a virtual circuit of a VDB, it does not, and cannot, show transmitting such instructions.

For the reasons given, it is submitted that the Examiner's rejection of claims 8, 12, 18, 25 and 29 under 35 U.S.C. §103(a) as being unpatentable Subramanian is not sustainable. Accordingly, Appellants respectfully request reversal of the rejection.

D. The Examiner erred in rejecting claim 9 under 35 U.S.C. §103(a) as being unpatentable over Subramanian in view of Suzuki

Suzuki is directed to packet switching in which two or more logical channels are set up for a connection. "In the event of an abnormal condition in the first logical channel the message packets are re-routed to the second logical channel." (Suzuki, Abstract.) Suzuki teaches that the multiple logical channels are set up "in response to each call-setup control packet" and "[a]t the end of a call, all the virtual circuits are released by a call-clearing control packet" (Suzuki, column 3, lines 24-32).

Appellants respectfully submit that Suzuki does not cure the deficiencies in Subramanian because Suzuki does not teach or suggest a VCB. The attributes of the Suzuki invention do not teach or suggest several characteristics of a VCB. For example, Suzuki does not allow the multiple virtual circuits to go to different destinations. Also, Suzuki does not teach that the multiple logical circuits be pre-established before an individual call or persist beyond the duration of an individual call, as with a VCB. In fact, Suzuki teaches the opposite, that the multiple logical channels should be released upon completion of the call. Therefore, Suzuki does not teach or suggest a VCB.

Claim 9 depends from claim 1, which recites *inter alia*, "a virtual circuit bunch" which is not taught or suggested by either Subramanian for the reason given in section A, or Suzuki for the reasons given above. Because neither Subramanian or Suzuki teach or suggest every limitation of Appellants' claim, a rejection of claim 9 is improper.



In addition, claim 9 recites "retransmit" and "cell interleaving problem" which are not shown by the references. Suzuki mentions "heavy traffic" and "trouble" in the passages cited by the Examiner (Suzuki, column 1, lines 33-37), but does not specify "cell interleaving" as the trouble. Also, in the passages cited, the system described must "send a new call-setup packet again to reestablish a new virtual circuit" which teaches away from retransmitting the information to an existing virtual circuit in the VCB.

For the reasons given, the Examiner's rejection of claim 9 under 35 U.S.C. §103(a) as being unpatentable over Subramanian and Suzuki is not tenable. Accordingly, Appellants respectfully request reversal of rejection.

E. The Examiner erred in rejecting claims 14, 15, 24, and 28 under 35 U.S.C. §103(a) as being unpatentable over Subramanian in view of Fisk

Claim 14 indirectly depends from claim 1, which recites *inter alia*, "a virtual circuit bunch" which is not taught or suggested by either Subramanian or Fisk for the reasons given above in sections A and B respectively. Because neither Subramanian or Fisk teach or suggest every limitation of Appellants' claim, a rejection of claim 14 is improper.

Further, claim 14 depends from claim 13 and therefore includes all the limitations thereof. Claim 13, however was not rejected under 35 U.S.C. § 103 as being unpatentable over Subramanian in view of Fisk. On the contrary, claim 13 was rejected under 35 U.S.C. § 103 as being unpatentable over Subramanian in view of Hiller, the error of which will be discussed below in section F. Because the combination of Subramanian in view of Fisk fail to teach or suggest every limitation of Appellants' claim 13, a rejection of claim 14 over such a combination is improper.

Claim 15 directly depends from claim 1, which recites *inter alia*, "a virtual circuit bunch" which is not taught or suggested by either Subramanian or Fisk for the reasons given above in sections

A and B respectively. Because neither Subramanian or Fisk teach or suggest every limitation of Appellants' claim, a rejection of claim 15 is improper.

Independent computer program product claim 24 recites, inter alia, "instructions for . . . aggregating those virtual circuits into a virtual circuit bunch." Again, because neither Subramanian or Fisk teach or suggest a virtual circuit bunch, neither Subramanian or Fisk teach or suggest every limitation of Appellants' claim. As such, a rejection of claim 24 is improper.

— Claim 28 directly depends from claim 24, and therefore includes all the limitations thereof. Because neither Subramanian or Fisk teach or suggest every limitation of Appellants' claim, a rejection of claim 28 is improper.

For the reasons given, it is submitted that the Examiner's rejection of claims 14, 15, 24 and 28 under 35 U.S.C. §103(a) as being unpatentable over Subramanian in view of Fisk is not tenable. Accordingly, Appellants respectfully request reversal of the rejection.

F. The Examiner erred in rejecting claims 13, 26 and 30 under 35 U.S.C. §103(a) as being unpatentable Subramanian in view of Hiller

Hiller is directed to allocating data from several telecommunication calls into each cell transmitted over a virtual circuit. Hiller mentions "permanent virtual circuits (PVC)," and "a plurality of . . . PVCs is provisioned", and "only PVCs which have been activated can carry the signals for telecommunications calls" (Hiller, column 2, lines 18-28). However, this disclosure is not the virtual circuit bunch (VCB) of Appellants' invention. Therefore Hiller does not cure the deficiencies in Subramanian.

Hiller does not teach how the PVCs are first set up or eventually broken down, or which processors or switches maintain a list of PVCs and their activity. Hiller only states that the "ingress node signals the egress node . . . the identification of the PVC" (Hiller, column 4, lines 21-27). Thus

there is no showing that the PVCs of Hiller are not merely the virtual circuits and virtual circuit paths of the prior art, requested the same way, set up in separate request messages, and managed the same way by the network. Hiller does not show that the PVCs are set up by a controller configured to set up a group of virtual circuits as a virtual circuit bunch.

Also, there is only one such group. For example, if all PVCs are full, none are added, and the "system busy" condition occurs (Hiller, column 10, lines 23-26, and Figure 15, item 1216). Because there is only one group, there is no need for names of the group. Thus there is no "hierarchy of aliases . . . utilized to conveniently refer to portions of . . . a virtual circuit bunch," (specification, page 23, lines 17-20).

Only the source terminal knows how many PVCs have been set up and whether they are all busy or not. Only the source terminal applies the methods of FIGs, 15 and 16 of Hiller, cited by the Examiner. The switches in the rest of the network are not shown to treat the PVCs as a group. Thus a VCB as that term is defined in Appellants' specification is not taught or suggested by Hiller.

Claim 13 directly depends from claim 1, which recites *inter alia*, "a virtual circuit bunch" which is not taught or suggested by either Subramanian as described in section A, or Hiller for the reasons given above. Because neither Subramanian or Hiller teach or suggest every limitation of Appellants' claim, a rejection of claim 13 is improper.

In addition, claim 13 recites "said request . . . specifies a plurality of destinations" which are not shown by the references. Nowhere do the references disclose a virtual path with virtual circuits going to different destinations. The Examiner points to a passage in Hiller which recites "receive path request" (Hiller, Fig. 15, item 1200). The cited passage does not disclose a single request for a path that specifies a plurality of destinations. A conventional virtual path includes separate virtual circuits that share the same destination (e.g., Hiller, column 2, lines 34-36). When such a path is requested,

plural destinations can not be specified. Making such a modification would change the principal of operation of the Hiller invention.

Independent computer program product claim 26 recites *inter alia*, "instructions for allocating a virtual circuit to all nodes participating in a multicast using a virtual circuit bunch." As described above, neither Subramanian nor Hiller teaches a virtual circuit bunch. Because neither Subramanian nor Hiller teaches or suggests every limitation of Appellants' claim, a rejection of claim 26 is improper.

Independent computer program product claim 26 recites "multicast" and "a virtual circuit bunch" which are not taught or suggested by the references. Neither Suzuki nor Subramanian nor Hiller teaches a VCB for the reasons given above. Furthermore, the Examiner does not show or even address where the references teach a multicast, or allocating a virtual circuit to all nodes participating in a multicast. Therefore a rejection of claim 26 under 35 U.S.C. §103 is improper. Claim 30 depends from claim 26, and is allowable for at least the same reason.

Claim 30 directly depends from claim 26, which recites *inter alia*, "a virtual circuit bunch" which is not taught or suggested by either Subramanian or Hiller for the reasons given above. Because neither Subramanian nor Hiller teaches or suggests every limitation of Appellants' claim, a rejection of claim 30 is improper.

In addition, claim 30 recites "transmitting said instructions . . . to a destination over a communications interface" which is not shown in the references. The Examiner does not address whether the references show that instructions are transmitted to a destination over a communications interface. Furthermore, since the references do not show instructions for allocating a virtual circuit in a multicast using a VDB, they do not, and can not, show transmitting such instructions.

For the reasons given, it is submitted that the Examiner's rejection of claims 13, 26 and 30 under 35 U.S.C. §103(a) as being unpatentable over Subramanian in view of Hiller is untenable. Accordingly, Appellants respectfully request reversal of the rejection.

CONCLUSION

The Examiner has failed to show that the independent claims are unpatentable over the cited references and failed to make a prima facie case of obviousness against the dependent claims. Each of the claims contains limitations not shown or fairly suggested by the prior art, and each achieves benefits not found in the prior art. Accordingly, Appellant respectfully requests that the Board of Patent Appeals and Interferences reverse the Examiner's rejections.

Respectfully submitted,

MCDERMOTT, WILL & EMERY



Thomas D. Robbins

Registration No. 43,369

600 13<sup>th</sup> Street, N.W.  
Washington, DC 20005-3096  
(202) 756-8000 DLS:TDR:daf  
**Date: November 15, 2000**  
Facsimile: (202) 756-8087

APPENDIX

1. A switching node, comprising:

- a. a switching matrix, and
- b. a controller to control said switching matrix, said controller configured to set up at least

one group of virtual circuits to respective one or more destinations as a virtual circuit bunch.

2. The switching node of claim 1 in which said switching node is an ATM switch.

3. The switching node of claim 1 in which said controller is configured to assign digital information from a source to one of a plurality of virtual circuits of a virtual circuit bunch.

4. The switching node of claim 6 in which the assignment of digital information from a source to one of a plurality of virtual circuits of a virtual circuit bunch is done without assigning said one of a plurality of virtual circuits to a connection.

5. The switching node of claim 1 in which the virtual circuits of a virtual circuit bunch going to a single destination may be routed over different paths.

6. The switching node of claim 1 in which the controller is configured to retransmit digital data from an assigned virtual circuit identifier (VCI) to an alternative VCI of the same or different port going to the same destination when a cell interleaving problem occurs.

10. Computer apparatus for connection to a switching node comprising:

- a. a bus;
- b. an input device, connected to said bus;
- c. a communications interface connected to said bus;
- d. a processor, connected to said bus, said processor configured to receive an input from a

user over said input device and to generate a single request to said switching node to establish a plurality of virtual circuits to respective one or more destinations as a virtual circuit bunch.

11. In a digital switching network having a plurality of interconnected nodes, a method of allocating virtual circuits, comprising the step of:

- a. providing an element for performing the step of establishing a plurality of virtual circuits from one node to at least one other node as a virtual circuit bunch in response to a single request.

12. The method of claim 11, in which the step of establishing a plurality of virtual circuits from one node to at least one other node as a virtual circuit bunch includes setting up switching tables when at least one subsequent node has acknowledged the request.

13. The method of claim 11, in which said request specifies a plurality of destinations.

14. The method of claim 13, in which said request also specifies the number of virtual circuits to be established to each destination.

~~15.~~ The method of claim 11, in which the request specifies the level of service to be provided by one or more virtual circuits.

~~16.~~ The method of claim 11, further comprising the step of establishing an end to end virtual circuit using at least one virtual circuit of said virtual circuit bunch.

~~17.~~ A method of allocating virtual circuits in a switching system, comprising the steps of:

- a. identifying virtual circuits at a node going to a common destination node; and
- b. aggregating those virtual circuits into a virtual circuit bunch.

~~18.~~ A method of providing a fast connect service in a digital switching network, comprising the step of:

- a. assigning a packet to a virtual circuit of a virtual circuit bunch.

~~19.~~ The method of claim 18 in which a packet is assigned a VCI without setting up a connection.

~~21.~~ A system for the transmission of digital communications, comprising:

- a. a plurality of user communication devices;
- b. a plurality of at least partially interconnected switching nodes, each node serviced by a node controller, servicing said user communications devices;
- c. in which at least one of said node controllers is configured to set up a group of virtual circuits to respective one or more destinations as a virtual circuit bunch.



~~22.~~ The system of claim 21 in which a virtual circuit from a user at one node is connected to a user at a destination node using a virtual circuit from said virtual circuit bunch.

~~23.~~ A computer program product, comprising:

- a. a memory medium, and
- b. a computer program stored on said memory medium, said computer program comprising instructions for establishing a plurality of virtual circuits from one node to at least one other node of a switching network as a virtual circuit bunch.

~~24.~~ A computer program product, comprising:

- a. a memory medium, and
- b. a computer program stored on said memory medium, said computer program comprising instructions for identifying virtual circuits at a node going to a common destination node; and for aggregating those virtual circuits into a virtual circuit bunch.

~~25.~~ A computer program product, comprising:

- a. a memory medium, and
- b. a computer program stored on said memory medium, said computer program comprising instructions for assigning a packet to a virtual circuit of a virtual circuit bunch.

26. A computer program product, comprising:

- a. a memory medium, and

b. a computer program stored on said memory medium, said computer program comprising instructions for allocating a virtual circuit to all nodes participating in a multicast using a virtual circuit bunch.

~~27.~~ A method of transferring the computer program of claim 23, comprising the step of transmitting said instructions from said memory medium to a destination over a communications interface.

~~28.~~ A method of transferring the computer program of claim 24, comprising the step of transmitting said instructions from said memory medium to a destination over a communications interface.

~~29.~~ A method of transferring the computer program of claim 25, comprising the step of transmitting said instructions from said memory medium to a destination over a communications interface.

30. A method of transferring the computer program of claim 26, comprising the step of transmitting said instructions from said memory medium to a destination over a communications interface.